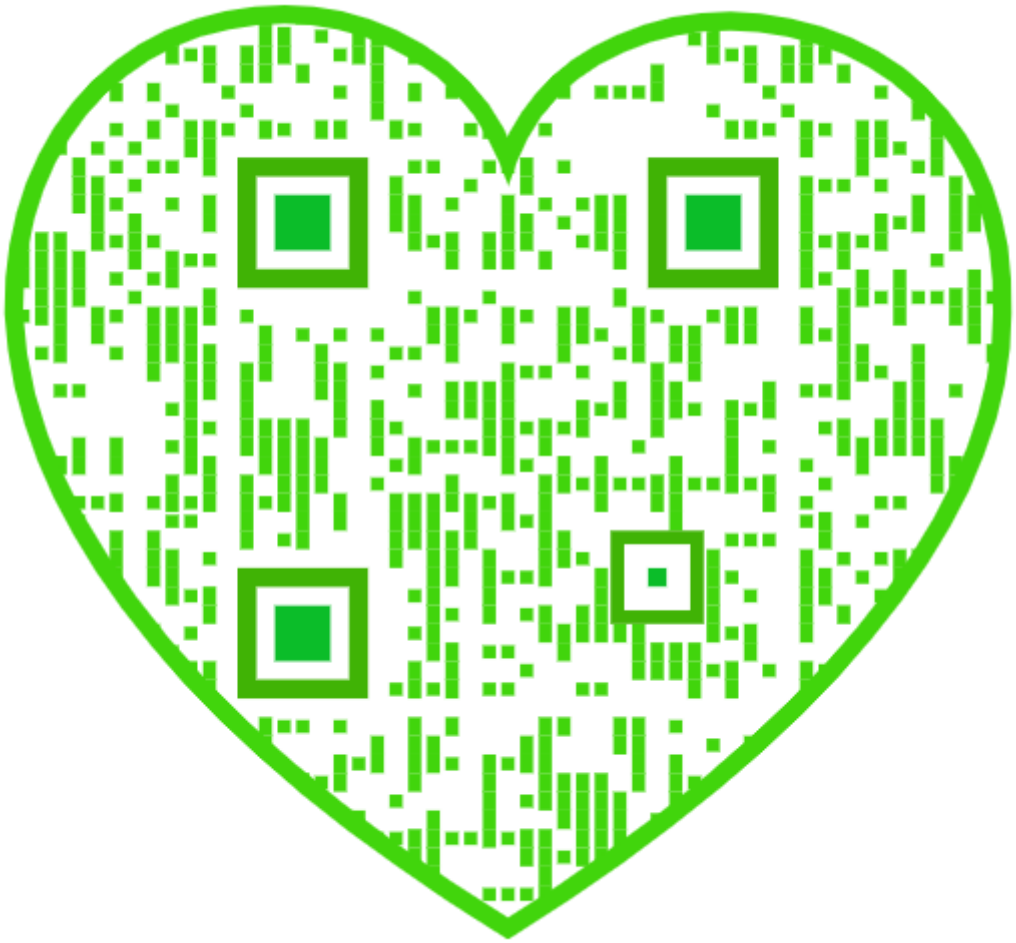
Master in Artificial Intelligence

Deployment I



Purpose

The purpose of the section is to help you learn how to deploy trained models into production environments to become a Successful Artificial Intelligence (AI) Engineer

At the end of this lecture, you will learn the following

- **How to deploy trained models into production environments, ensuring they integrate smoothly with existing systems and meet performance requirements**

How to deploy trained models into production environments

Trained Model

+

Existing Systems

Meet Performance Requirements

Model Serialization

ONNX

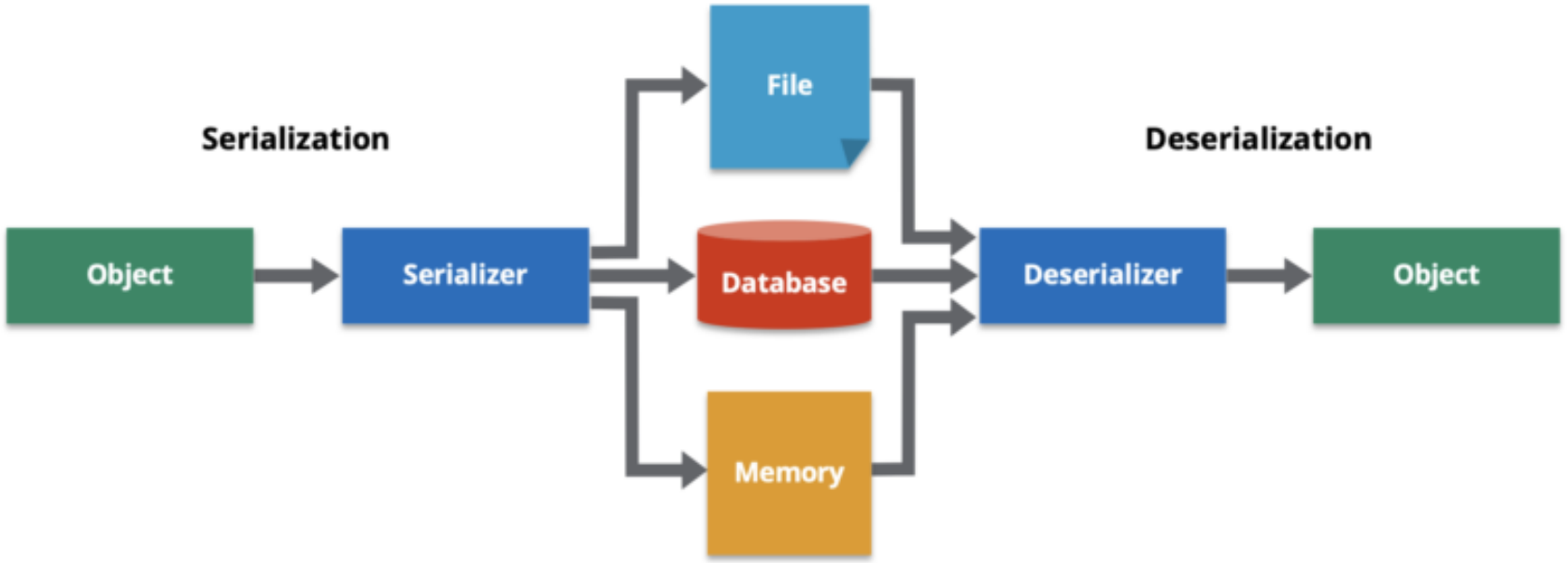
Joblib

What is Model Serialization

ONNX

Joblib

What is Model Serialization



How to serialize the trained model using Common serialization formats like pickle, joblib, or ONNX

ONNX

Joblib

How to serialize the trained model using pickle

1. **Using Pickle:**

```python
import pickle

# Assuming 'model' is your trained machine learning model
with open('model.pkl', 'wb') as f:
    pickle.dump(model, f)
```

To load the model:

```python
with open('model.pkl', 'rb') as f:
    loaded_model = pickle.load(f)
```

How to serialize the trained model using joblib

1. **Using Joblib:**

python

```python
from joblib import dump, load

# Assuming 'model' is your trained machine learning model
dump(model, 'model.joblib')
```

To load the model:

python

```python
loaded_model = load('model.joblib')
```

How to serialize the trained model using ONNX

1. **Using ONNX:**

```python
import onnx
from onnxruntime import InferenceSession

# Assuming 'model' is your trained machine learning model
onnx_model = onnx.load("model.onnx")
onnx.checker.check_model(onnx_model)
session = InferenceSession(onnx_model.SerializeToString())
```

To load the model:

```python
# Once the session is created, it's ready to make predictions
```

How to choosing a serialization format

ONNX

Joblib

How to choosing a serialization format

Pickle

- Can handle most Python objects, including custom classes

Pickle

- Not always the most efficient for large NumPy arrays or models with C extensions

Joblib

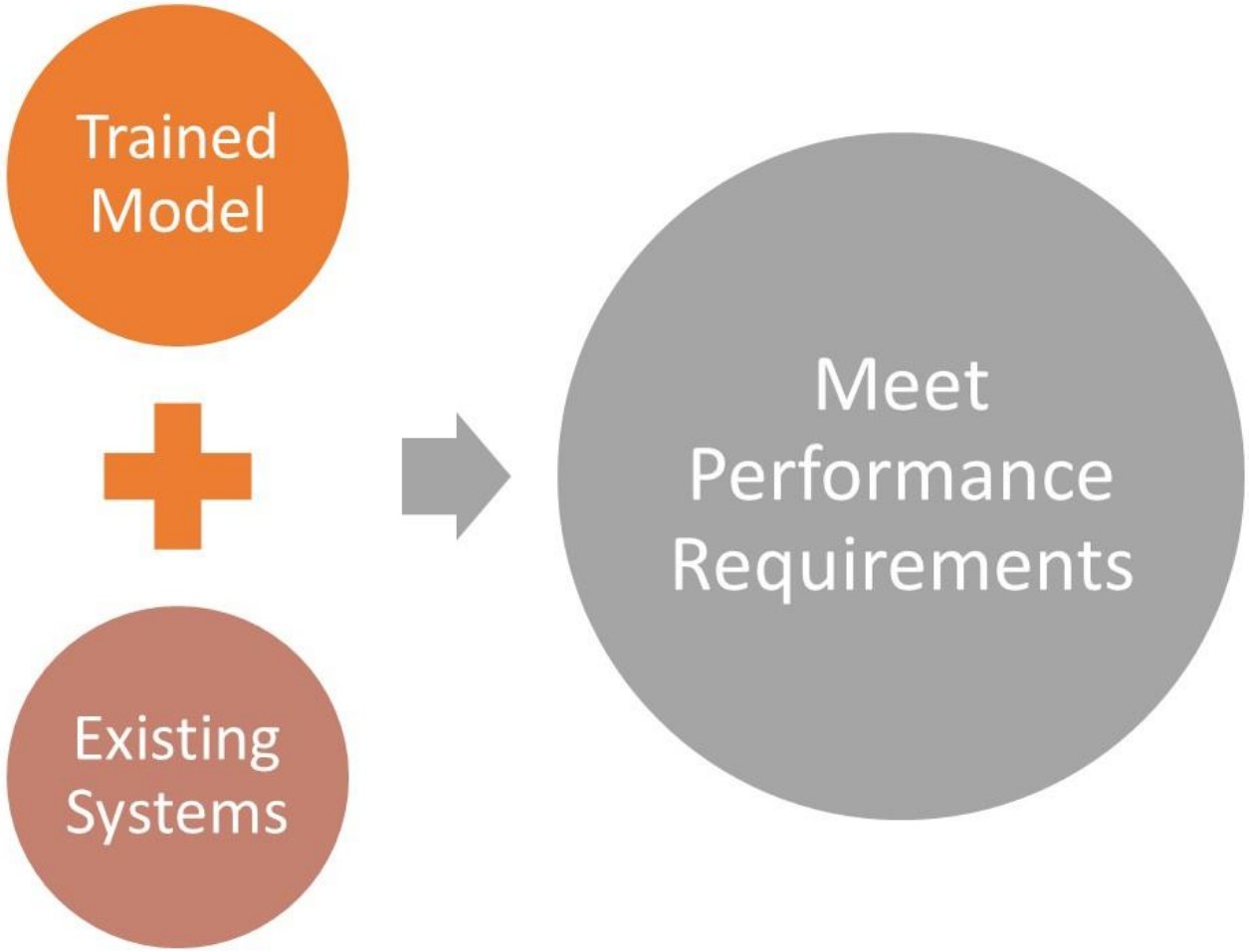
- More efficient for storing large NumPy arrays
- Commonly used in the scikit-learn ecosystem

ONNX (Open Neural Network Exchange)

- Can be run on various platforms and frameworks using the ONNX Runtime
- Particularly useful for interoperability between different deep learning frameworks

What is next?

How to deploy trained models into production environments



Master in Artificial Intelligence

Deployment I